Inference Trees:
Adaptive Inference with Exploration

Tom Rainforth
Joint work with Yuan Zhou, Xiaoyu Lu, Yee Whye Teh, Frank Wood, Hongseok Yang, and Jan Willem van de Meent
Overview

• Adaptive inference methods need to explore, not just exploit
• New class of inference algorithms — Inference Trees — based around Monte Carlo tree search
• Targeted exploration using density estimation of log sample weights
\[ \pi(x) = \frac{\gamma(x)}{\omega} \]
Background: Importance Sampling

\[ \gamma(x) \]
Background: Importance Sampling

\[ \gamma(x) \quad q_t(x) \]
Background: Importance Sampling

\[ \hat{\pi}(\cdot) = \frac{1}{N} \sum_{n=1}^{N} w_n \delta_{\hat{x}_n}(\cdot) \]

\[ \frac{1}{N} \sum_{n=1}^{N} w_n \]

\[ \hat{x}_n \sim q_t(x) \]

\[ w_n = \frac{\gamma(\hat{x}_n)}{q_t(\hat{x}_n)} \]
Background: Adaptive Monte Carlo Monte Carlo

\[ q_{t+1} = \text{argmin}_q \text{KL} (\hat{\pi} || q) \]
Background: Adaptive Monte Carlo

- $\gamma(x)$
- $q_t(x)$
- $q_{t+1}(x)$
Exploitation vs Exploration

- Exploitation: sample in regions where we think the posterior mass is high
- Exploration: sample in regions where our uncertainty is high
- Utility from samples originates not only from direct contribution to estimator, but also information provided for future sampling
we reach a leaf. At a high level, we want to allocate resources to
of the key underlying themes motivating ITs.

Though there are a range of approaches to adapting the
inference or integration. They require a base algorithm which is
adaptively applied to different regions of the target space.

Characterization of the tree learning algorithm. Each
Traversal
Refinement
Propagation
Background and Related Work

The simplest
The most
The initial
The key

In the stratified sampling setting (\(Q\)), one starts
by running inference directly and updating the local estimate,
or splitting the node to expand the tree and running inference at
the root note and recursively choosing the left or right node until
that the region of a parent node is the union of its children and
areas where the posterior density or our uncertainty is high. In
the root note and recursively choosing the left or right node until
that the region of a parent node is the union of its children and
areas where the posterior density or our uncertainty is high. In

\[ \hat{p} = \max \{ \text{expected reward} \} \]

Though there are a range of approaches to adapting the
inference or integration. They require a base algorithm which is
adaptively applied to different regions of the target space.

\[ \hat{p} = \max \{ \text{expected reward} \} \]

In the stratified sampling setting (\(Q\)), one starts
by running inference directly and updating the local estimate,
or splitting the node to expand the tree and running inference at
the root note and recursively choosing the left or right node until
that the region of a parent node is the union of its children and
areas where the posterior density or our uncertainty is high. In

\[ \hat{p} = \max \{ \text{expected reward} \} \]

In the stratified sampling setting (\(Q\)), one starts
by running inference directly and updating the local estimate,
or splitting the node to expand the tree and running inference at
the root note and recursively choosing the left or right node until
that the region of a parent node is the union of its children and
areas where the posterior density or our uncertainty is high. In

\[ \hat{p} = \max \{ \text{expected reward} \} \]
We will for now assume that we are using ITs for inference. We recursively update the tree with the new estimates to improve the performance or integration. They require a base algorithm which is used to either split the node to expand the tree and run inference at the root node and recursively choose the left or right node until the union of all leaf nodes is the full space. In the Background and Related Work section, we discuss some metrics that can substantially increase the effectiveness of ITs as it enables the use of inventive uncertainty or splitting the node to expand the tree and run inference at the root node and recursively choose the left or right node until the union of all leaf nodes is the full space. In the stratified sampling setting, we reach a leaf. At a high level, we want to allocate resources to allocate computational resources, starting at the exploration. For example, we introduce a novel method to adaptively allocate computational resources, starting at the proposal that minimizes the exploration. Here the proposal chooses the form of the MC scheme that returns weighted samples and makes the exploration. We now discuss some traversal methods where the region of a parent node is the union of its children and then look to optimally allocate computational resources at each iteration, choose the proposal that minimizes the exploration. Though there are a range of approaches to adapting the reward and the agent's goal is to maximize the long term reward and the agent's goal is to maximize the long term.

Multi-Armed Bandits

One common strategy is upper confidence bounding (UCB) (4). One common strategy is upper confidence bounding (UCB) (4). Here one starts with a number of strata, i.e. partitions, for a target estimator (4). One common strategy is upper confidence bounding (UCB) (4). This leads to an explicit exploration. Here the proposal chooses the form of the MC scheme that returns weighted samples and makes the exploration.

Inference Trees

Inference Trees: Adaptive Inference with Exploration

Tom Rainforth
\[
\hat{\gamma}(\cdot) = \hat{\gamma}_1(\cdot) + \hat{\gamma}_2(\cdot) + \hat{\gamma}_3(\cdot)
\]

\[
\hat{\gamma}_1(\cdot) \\
N_1 = 180
\]

\[
\hat{\gamma}_2(\cdot) \\
N_2 = 800
\]

\[
\hat{\gamma}_3(\cdot) \\
N_3 = 20
\]
Why Split and Control?

- We no longer need to draw iid samples
  - More explicit control for resource allocation
  - Can gather additional information
- Easy to maintain consistency under adaptation
Inference Trees
Traversals
Inference Trees: Adaptive Inference with Exploration
Tom Rainforth

Propagation

0

1

2

3

4

5

6
Inference Tree Estimator / Propagation

\[ \hat{\omega}_j = \frac{1}{N_j} \sum_{n=1}^{N_j} w_j^n \]
Inference Tree Estimator / Propagation

\[ \hat{\omega}_j = c_j (\hat{\omega}_{l_j} + \hat{\omega}_{r_j}) + (1 - c_j) \frac{1}{N_j} \sum_{n=1}^{N_j} w^n_{j} \]

\[ \hat{\gamma}_j (\cdot) = c_j (\hat{\gamma}_{l_j} (\cdot) + \hat{\gamma}_{r_j} (\cdot)) + (1 - c_j) \frac{1}{N_j} \sum_{n=1}^{N_j} w^n_{j} \delta_{\hat{x}_n^j} (\cdot) \]

\[ \hat{\pi}_0 (\cdot) = \frac{\hat{\gamma}_0}{\hat{\omega}_0} \]
Consistent Regardless of Traversal and Refinement Strategies:

**Theorem 1.**

\( \hat{\pi}_0(\cdot) \) converges weakly to \( \pi(x) \).
What is the optimal allocation of samples?

One might expect we should sample in proportion to the marginal likelihood, but in fact we have

$$\tau_j = \sqrt{\omega_j^2 + (1 + \kappa)\sigma_j^2}$$

Marginal likelihood Squared

Smoothness parameter

Variance of weights

$$\kappa \geq 0$$
Traversal
Multi Armed Bandits for Stratified Sampling

UCB: at each round, choose the arm $j$ that maximises

$$u_j = \frac{1}{M_j} \left( \hat{r}_j + \frac{\beta \log \sum_i M_i}{\sqrt{M_j}} \right)$$

**Estimated Reward**

**Optimism Boost**

[Carpentier et al 2015]
Reward

\[ \hat{r}_j = \hat{r}_j \]
Reward

\[ \hat{r}_j = (1 - \delta) \hat{r}_j + \delta \hat{p}_j^s \]

Exploitation | Targeted Exploration
Reward

\[ \hat{r}_j = (1 - \delta) \frac{\hat{T}_j}{\hat{T}_{pa}(j)} + \delta \frac{\hat{p}^s_j}{\hat{p}^s_{pa}(j)} \]

- **Exploitation**
- **Targeted Exploration**
How can we perform targeted exploration?

Is $\omega_j > w_{th}$?
Density estimation of the log weights

Is $\omega_j > w_{th}$?
Density estimation of the log weights

Will $\hat{\omega}_j(T)$ be larger than $w_{th}$?
Density estimation of the log weights

\[
P(\hat{\omega}_j(T) > w_{th}) \approx P(\max(\omega_j^{1:T}) > w_{th}) \approx 1 - \Psi(\log w_{th})^T
\]
Inference Tree Traversal Target

\[ u_j = \frac{1}{M_j} \left( (1 - \delta) \frac{\hat{r}_j}{\hat{r}_{\text{pa}(j)}} + \delta \frac{\hat{p}^s_j}{\hat{p}^s_{\text{pa}(j)}} + \frac{\beta \log \sum_i M_i}{\sqrt{M_j}} \right) \]

- **Exploitation**
- **Targeted Exploration**
Another important point of interest is that it is perfectly permissible for A high level description of this process is shown below.

As explained in the main paper, effectively partitioning in the space of pre-image is in Because the distribution over can think in terms of performing inference on such that we can sequentially generate through ensure that then

In general, A and [2] the hierarchical partitioning of any region can be defined using a uniform hypercube, the probability of generating an is just the hypervolume of 

{ } and the leaf nodes on the target space

The situation is just the (known) area of where we note that the partition between

Numbering left to right, [1] shows the original proposal [2] shows the partitioning implied by [3] shows the partitioning implied by imposed by the tree. [3] shows the partitioning implied by

Let the leaf nodes on the space of the proposal factorize as 

\[ q(x \{ x \in A_j \}) = \frac{q(x) \mathbb{I}(x \in A_j)}{\|B_j\|} \]
Refinement
Split Criterion

- Prefer splits which lead to concentration of probability mass to few nodes

\[
\text{LOSS(split)} = \hat{\omega}_l \log \frac{\|B_l\|}{\hat{\omega}_l} + \hat{\omega}_r \log \frac{\|B_r\|}{\hat{\omega}_r}
\]

- Sample a large number of candidate splits at random
- Choose candidate split which minimises loss criterion for existing samples
Inference Trees Recap

Traversal

Refinement

Propagation
Experiments: Gaussian Mixture Model

(a) PI-MAIS  (b) Naïve IT  (c) IT
Experiments: Gaussian Mixture Model

![Graph showing ESS vs Number of Samples for various methods]

- IT
- IS
- Naive IT
- PI-MAIS
Experiments: Chaotic Dynamics Model

Inference Trees: Adaptive Inference with Exploration
Tom Rainforth

Figure 3: Kernel density estimate of projected posterior estimates for the GMM (a-c) and the chaos model (d-f). We use a linear projection of the original 8/4-dimensional spaces and exaggerate the variance of the modes for visualization purposes. For both problems, the IT has successfully recovered all modes and inferred that all the modes have equal mass. Though the naïve IT implementation produced good estimates for the modes it found, it missed modes for both problems. For the GMM, PI-MAIS found a number of modes but still missed some and misestimated their relative masses. For the chaos model, PMMH only found a single mode.

Figure 4: Convergence of log ML and ESS for chaos model, conventions as per Figure 2. PMMH is not shown as it returns unweighted samples and no ML estimate; other results are given in Appendix I.

Figure 5: Inference Trees: Adaptive Inference with Exploration. We illustrate the performance of ITs in (a) a GMM and (b) a chaotic dynamics model. The IT process allows information to be gathered even in the face of degeneracy. Constraining different sweeps to different regions allows samples to be “forced through” the resampling steps, hereby dealing with long-range dependencies. This is done without losing the key benefits of SMC, as gains from resampling are still seen when running inference within a particular region. Note that ITs only require an unbiased estimate for the weights in a manner akin to pseudo-marginal methods [2].

To test ITs in this setting, we consider an adaptation of the chaotic dynamical system tracking problem introduced by [31], details for which are given in Appendix H. The model comprises of an extended Kalman filter where we have dynamics parameters $\theta$, latents $x_1:T$, and observations $y_1:T$. We desire to conduct inference over both the dynamics parameters and the latent variables, but will only use ITs to control the sampling of the former. This model contains long-range dependencies because the dynamics parameters affect each transition and so the smoothing marginal $p(\theta | y_1:T)$ is very different to the filtering marginal $p(\theta | y_1)$. In fact, the two are so different that using the so-called one-step-optimal proposal, the target for most methods of SMC proposal adaptation [21], provides no noticeable performance improvement over simply sampling from the prior.

Because PI-MAIS requires an MCMC sampler to be run on the target $p(\theta | y_1:T)$, it is inappropriate for this problem. We instead compare to using SMC without adaptation, SMC with 1000 times more particles, the naïve IT implementation, and PMMH [4], a method explicitly designed for dealing with global parameters in SMC. We allowed a budget of $1 \times 10^7$ target evaluations and used 8 SMC sweeps of 500 particles per refinement step for the IT approaches. Details on parameters setups are given in Appendix H. We used the same comparison metrics as for the GMM, with results shown in Figures 3 and 4. We see that ITs again outperformed the other methods.

Conclusions
We have introduced inference trees (ITs), a new adaptive inference algorithm drawing on ideas from Monte Carlo tree search. We have shown that, by carrying out explicit exploration in the adaptation process, ITs can avoid common pathologies with other adaptive schemes and reliably uncover multiple modes. We have consequently found that, for the tested models, ITs outperformed previous state-of-the-art adaptive importance sampling and particle MCMC methods. In addition to the immediate utility of the proposed approach, we believe that the general IT framework opens up many opportunities for new research, due to the separation between their consistency and the specifics of the learning algorithm. For example, ITs can also be used for integration (see Appendix J).
Experiments: Chaotic Dynamics Model

(d) PMMH  (e) Naïve IT  (f) IT
Recap

• Adaptive inference methods need to explore, not just exploit
• New class of inference algorithms — Inference Trees — that can outperform current adaptive importance sampling and SMC methods
• Targeted Exploration: new estimator for predicting presence of significant probability mass using poor quality samples
Thanks!


rainforth@stats.ox.ac.uk

http://www.robots.ox.ac.uk/~twgr/